

# Brief Announcement: The Impact of Classical Electronics Constraints on a Solid-State Logical Qubit Memory

James E. Levy, Anand Ganti, Cynthia A. Phillips, Benjamin R. Hamlet,  
Andrew J. Landahl<sup>\*</sup>, Thomas M. Gurrieri, Robert D. Carr, and Malcolm S. Carroll<sup>†</sup>  
Sandia National Laboratories<sup>†</sup>  
Albuquerque, NM, USA  
{jlevy, aganti, caphill, brhamle, mscarro, alandahl, tmgurri, rdcarr}@sandia.gov

## ABSTRACT

We present and analyze an architecture for a logical qubit memory that is tolerant of faults in the processing of silicon double quantum dot (DQD) qubits. A highlight of our analysis is an in-depth consideration of the constraints faced when integrating DQDs with classical control electronics.

**Categories and Subject Descriptors:** B.3.4 [Hardware]: Memory Structures: reliability, testing, and fault-tolerance; C.C.m [Computer Systems Organization]: Miscellaneous

**General Terms:** Algorithms, Design, Reliability

## 1. INTRODUCTION

Quantum computers promise algorithmic speedups for numerous computational problems, yet they are extremely sensitive to noise [10]. Fortunately, the *threshold theorem* for fault-tolerant quantum computation guarantees that arbitrarily reliable quantum computation is possible if all operations in a quantum circuit fail with a probability below an *accuracy threshold* [4] that depends on the architecture and noise model. We encode a *logical qubit* in multiple physical qubits. In this Brief Announcement, we compute the accuracy threshold for an architecture that implements the first level of fault-tolerant quantum error correction of a single encoded logical qubit. Unlike previous studies which consider some of the impacts of specific implementation constraints [3, 9, 5], we explore the impact of integration of Si DQD qubits with their classical control electronics.

Since quantum information transport between Si DQDs is problematic, we focus on local-check error correction codes, where quantum information is manipulated locally. We describe a layout of the supporting electronics and qubits for a code capable of detecting one error. This layout considers physical qubit geometry, electronics constraints, thermal staging (i. e. determining the temperature for components), and routing. Using integer programming, we found an op-

timal schedule of error-correction qubit operations. Parallel Monte Carlo simulation shows that there is still too much idle time: there is no finite error probability for the native gates such that the error-correction system would be beneficial. However, by adding dynamic decoupling, a series of timed pulses that can reverse some errors, there may be a threshold.

## 2. PHYSICAL QUBIT CONSTRAINTS

Silicon DQD qubits have a number of features that are appealing for quantum computing. However, they face certain constraints that make developing a fault-tolerant logical qubit challenging. We summarize some of these here; See our full paper [8] for more details.

Si DQD qubits lie on a 2D complementary metal-oxide-semiconductor (CMOS) substrate that must be cooled to roughly 100 mK. Each qubit requires multiple electrical control lines for tuning and gating; our design uses 17 per qubit. Qubits cannot currently be transported reliably, even using quantum “software” solutions like teleportation. Using standard notation [10], Table 1 lists a set of hypothetical nearest-neighbor quantum gates with their gate times and failure rates. These gates, analogous to those developed for GaAs DQD qubits [14], are sufficient to perform quantum error correction.

The dominant failure mode for the non-identity gates is assumed to come from charge fluctuations that lead to unknown shifts in the exchange energy of the Si DQD qubit [6]. The dominant failure mode for the identity gates comes from non-uniform external magnetic fields [15]. In related GaAs DQD qubit technology, such fields cause each qubit to decohere with a fidelity half-life of  $T_2^* \sim 3$  ns [11]. In an open-loop control scheme called *dynamical decoupling* (DD) [7], any idle time is contained in a gate sequence *Z-Idle-Z-Idle*, where Idle is a set of  $k \geq 0$  idle ticks. With DD, this decoherence time can be extended to an increased fidelity half-life of  $T_2 \sim 1$   $\mu$ s [11]. While Si DQD qubits have yet to be demonstrated experimentally, spin ensemble studies of electrons confined by donors in Si suggest that  $T_2^* \sim 1$   $\mu$ s and  $T_2 \sim 60$  ms for Si DQD qubits [16], which is considerably longer. It is not *prima facie* clear that DD will be beneficial for Si DQD qubits because the gates used in the DD sequence are themselves flawed. We therefore consider architectures both with and without DD sequences; for technical ease, we actually consider the DD sequence *X-Idle-X-Idle*.

<sup>\*</sup>This work was performed while this author was at the University of New Mexico

<sup>†</sup>Sandia is a multipurpose laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

$M_Z,  1\rangle$	$S$	$Z$	$X_{\pi/2}$	$X, CPHASE$	$I^*$	$I$
$\tau$	$\tau$	$2\tau$	$3\tau$	$4\tau$	$\tau$	$\tau$
$p/30$	$p$	$2p$	$4p$	$4p$	$10^{-2}$	$10^{-6}/2$

**Table 1: Gate Times and Failure Probabilities.**  $\tau = 30$  ns.  $p$  is variable, but estimated to be 0.3% with current technology.  $I^*$  denotes the bare identity gate and  $I$  denotes the DD-protected identity gate.

### 3. THE BACON-SHOR CODE

We consider storing a logical qubit in the Bacon-Shor code [2], which encodes one logical qubit in nine physical data qubits and uses 12 ancillary *syndrome* qubits to store information about where errors have occurred. When laid out in 2D as depicted in Fig. 1, the syndrome qubits are adjacent to the data qubits they need to interact with, obviating the need for quantum transport. Because measuring syndrome qubits could be faulty, we repeat each syndrome qubit measurement and act on the information only if both measurements agree. Rather than correcting errors using possibly faulty gates, we log information about the errors using classical circuitry and apply a net correction after a final readout of the logical qubit.

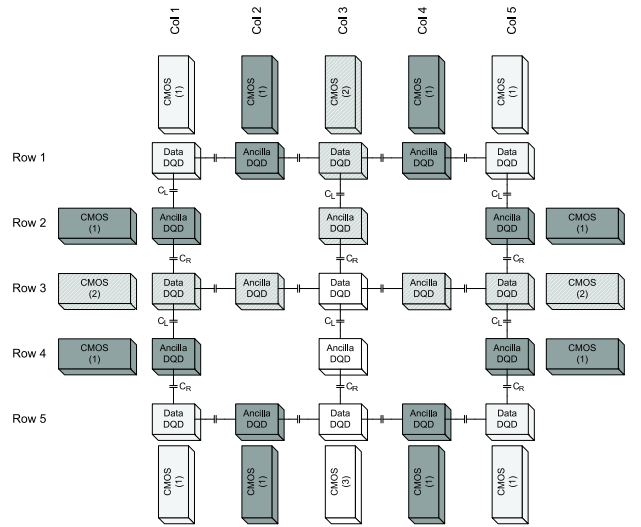
### 4. ELECTRONICS CONSTRAINTS

The cooling power of modern dilution refrigerators at 100 mK cannot keep up with power-hungry control electronics, so we thermally stage the electronics. Cooling power also limits the number of signal lines allowed between cryostat stages, requiring multiplexing. This limits the parallelism of quantum circuits. In our design, multiplexers, demultiplexers, and static memory elements are at 100 mK, digital readout circuitry is at 4 K to reduce capacitance from long wires into the cryostat, and the remaining circuitry (master CPU and pulse generators) are at 300 K. To minimize the number of signal lines needed, our design uses one pulse generator for each gate type. We conservatively estimate that the fastest clock rate sustainable which assures high timing precision (low jitter) is 30 ns, as listed in Table 1.

The proposed two-qubit gate requires a capacitive coupling between qubits [14]. Each pair of neighboring qubits can couple on either the left side or the right, denoted by  $C_L$  and  $C_R$  in Figure 1. The set of pulses sent to a qubit to perform a two-qubit gate with its left neighbor is different from those applied to perform the same gate with its right neighbor. The coupling uses a metal trace between the qubits which is susceptible to electronic cross-talk. We minimize cross-talk by packing qubits tightly. This restricts the number of routing channels—for a 65 nm process using minimum metal width and spacing, 8 metal layers,  $1 \mu\text{m}^2$  qubits, and  $0.5 \mu\text{m}$  spacing, classical controllers can communicate with at most 3 qubits. Each classical controller can send only one gate signal at a time to reduce errors from cross-talk; each qubit a controller controls can either perform the gate or remain idle. While our design minimizes numerous error sources while maximizing clock speed, it constrains the scheduling of the error correction operations.

### 5. SCHEDULING

We briefly describe the integer program (IP) used to compute the optimal schedule for the BS9(21) architecture de-



**Figure 1: Enclosed 21 Qubit Bacon-Shor Architecture.** Three types of CMOS control blocks, those that control: 1 DQD (light & dark gray), 2 DQD (Gray w/ stripes) and 3 DQD (white).

scribed in Sections 3 and 4 and shown in Fig. 1. We divide time into “ticks” of size  $\tau$ , the minimum gate length (see Table 1). All gate times are small multiples of  $\tau$ . We encode time in variable subscripts, allowing binary decision variables. The size of the formulation thus depends upon an upper bound for the schedule length, or *makespan*.

We must find a legal start tick for each row and column circuit operation for each qubit. For data qubits, row/column operations cannot interleave except that  $S$  and  $CPHASE$  operations at the boundary between row and column operations can commute. Thus we must always obey some precedence constraints within each circuit, and we must conditionally obey precedence constraints based on circuit ordering decisions for each data qubit. Qubits may be idle at any given tick. All qubits that share a controller may perform the operation the controller is signaling, and must execute gates without interruption. There are three ways a controller can signal a  $CPHASE$ : internal (both qubits share a controller), right dot, or left dot. Multiple qubits sharing a controller can simultaneously execute  $CPHASE$  gates with qubits controlled by other controllers, provided the coupling is of the appropriate (left/right) type. Each controller can measure at most one qubit at a time.

We wish to minimize the total idle time of all qubits. Data qubits are in continuous (re)use, so their idle time is determined by the makespan. Idle time for ancilla is only counted between the preparation and the measurement.

We first computed the minimum makespan with one IP. We then used the minimum makespan as the first estimate for the makespan when computing a minimum-idle-time schedule. This objective explicitly trades off makespan vs. ancilla idle time. The optimal schedule for this makespan had only two total idle ticks summed over all ancilla. Because increasing the makespan would add 9 ticks of idle time taken over all the data qubits, while possibly only removing 2 ticks from the ancilla, this schedule is optimal over all makespans.

## 6. ACCURACY THRESHOLD

We assessed our logical qubit architecture with Monte Carlo simulation using a self-modified extension of the QDNS simulator [12]. We assumed a *depolarized noise* (DPN) model in which each non-measurement gate works flawlessly with probability  $1 - p_{\text{gate}}$  and otherwise is followed by a non-identity Pauli operator [10] selected uniformly at random. The probabilities  $p_{\text{gate}}$  for each gate were drawn from Table 1 and varied with a simulation parameter  $p$ . Identity gates (idles) had  $p$  fixed at 0.3% as indicated in the table; we fixed these at 0.3% because there is no known mechanism for improving them beyond replacing the bare identity ( $I^*$ ) with a DD-protected identity ( $I$ ). We also considered an *unbiased* DPN model in which each nonidentity gate has  $p_{\text{gate}} = p$ .

We studied three measurement noise models, of increasing realism. In the first black-box model, two-qubit syndrome measurements occur in a single step, subject to DPN with  $p_{\text{gate}}$  values drawn from Table 1. In the second Steane model, two-qubit syndrome measurements use Steane’s fault-tolerant protocol [13], where each nonlocal gate is subjected to DPN, and in the third Si DQD model, two-qubit syndrome measurements use the quantum circuits in our implementation of the Bacon-Shor code, with each gate subject to DPN. In this last setting, we considered both an IP idle-minimized schedule without DD and a hand-generated schedule with *X-Idle-X-Idle* DD.

Our analysis uses a technical but standard *extended rectangle* method [1] on top of the Monte Carlo simulations to estimate the accuracy threshold. In our setting, this threshold is defined to be the maximum  $p$  such that the failure probability  $p_f$  of quantum error-correction is less than the failure probability of the most faulty gate. We used binary search on  $p$  to find the  $p$  for which  $p_f$  equals this fault probability. Table 2 summarizes our results for each noise model and various  $p_{\text{gate}}$  values for the measurement and idle gates.

	Black-box	Steane	no DD	DD
$p_{M_Z} = 0$	1700			
$p_{M_Z} = 2p/3$	1100			
$p_I = 10^{-2}$		73		
$p_I = 10^{-5}$		110		
$p_I = 0$		110		
unbiased			DNE	5.5
biased			DNE	2.0

**Table 2: Monte-Carlo accuracy threshold estimates ( $\times 10^{-5}$ ) for various syndrome measurement models. All gates are equally likely to fail unless otherwise specified. All values are  $\pm 1$  in the smallest nonzero significant figure. The specifics of the noise models are described in the text.**

As expected, the accuracy threshold shrinks as the measurement noise model becomes more demanding. The main surprise is that an accuracy threshold simply doesn’t exist when DD isn’t used, even for the schedule with provably minimum idle time. The gate counts in the optimal schedule, summarized in Table 3, show that idles dominate the schedule. Furthermore, we haven’t allowed the idle failure rates to scale with  $p$ . This causes  $p_f$  to be fixed near 0.2 in the non-DD case, even if all other gates are error-free. While the introduction of  $X$  and  $I$  gates in the DD scheme ostensibly increases  $p_f$ , it is more than compensated for by

the reduction of the fault rate of  $I^*$  to that of  $I$ , a factor of 2000 as shown in Table 1. Thus dynamical decoupling is *essential* for fault-tolerantly preserving a logical qubit in the Bacon-Shor code in our architecture for Si DQD qubits.

Gate	BS9(21) w/o DD	BS9(21) with DD
	# of Gates	# of Gates
Prep $ 1\rangle$	12	12
$X_{\pi/2}$	42	42
$Z_{\pi/2}$	18	18
$X$	0	104
<i>CPHASE</i>	24	24
$M_Z$	12	12
$I^*, I^*$	95	219

**Table 3: Gate count for realizing error correction using the native gate set in our architecture**

## 7. REFERENCES

- [1] Panos Aliferis, Daniel Gottesman, and John Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quant. Inf. Comput.*, 6(2):97–165, 2006.
- [2] Dave Bacon. Operator quantum error-correcting subsystems for self-correcting quantum memories. *Phys. Rev. A*, 73(1):012340, Jan. 2006.
- [3] S. Balensiefer, L. Kreger-Stickles, and M. Oskin. An evaluation framework and instruction set architecture for ion-trap based quantum micro-architectures. In *International Symposium on Computer Architecture*, 2005.
- [4] D. Gottesman. An introduction to quantum error correction and fault-tolerant quantum computation, 2009. arXiv:0904.2557.
- [5] L. C. L. Hollenberg, A. D. Greentree, A. G. Fowler, and C. J. Wellard. Two-dimensional architecture for donor-based quantum computing. *Physical Review B*, 74(045311), 2006.
- [6] X. Hu and S. Das Sarma. Charge-fluctuation-induced dephasing of exchange-coupled spin qubits. *Physical Review Letters*, 96:100501, 2006.
- [7] K. Khodjasteh and D. A. Lidar. Fault-tolerant quantum dynamical decoupling. *Physical Review Letters*, 95, 2005.
- [8] J. E. Levy, A. Ganti, C. A. Phillips, B. R. Hamlet, A. J. Landahl, T. M. Gurrieri, R. D. Carr, and M. S. Carroll. The impact of classical electronics constraints on a solid-state logical qubit memory, 2009. arXiv:0904.003.
- [9] D. Loss and D. P. DiVincenzo. Quantum computation with quantum dots. *Phys. Rev. A*, 57(1):120–126, Jan. 1998.
- [10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.
- [11] J. R. Petta, A. C. Johnson, J. M. Taylor, E. A. Laird, A. Yacoby, M. D. Lukin, C. M. Marcus, M. P. Hanson, and A. C. Gossard. Coherent manipulation of coupled electron spins in semiconductor quantum dots. *Science*, 309:2180–2184, 2005.
- [12] P. Abroonits S. Imre and D. Darabos. Quantum designer and network simulator. In *ACM International Conference on Computer Frontiers*, pages 89–95, April 2004.
- [13] A. M. Steane. Active stabilization, quantum computation, and quantum state synthesis. *Phys. Rev. Lett.*, 78(11):2252–2255, Mar. 1997.
- [14] J. M. Taylor, H.-A. Engel, W. Dur, A. Yacoby, C. M. Marcus, P. Zoller, and M. D. Lukin. Fault-tolerant architecture for quantum computation using electrically controlled semiconductor spins. *Nature*, 177, 2005.
- [15] J. M. Taylor, J. R. Petta, A. C. Johnson, A. Yacoby, C. M. Marcus, and M. D. Lukin. Relaxation, dephasing, and quantum control of electron spins in double quantum dots. *Phys. Rev. B*, 76:035315, 2007.
- [16] A. M. Tyryshkin, S. A. Lyon, A. V. Astashkin, and A. M. Raitsimring. Electron spin relaxation times of phosphorus donors in silicon. *Phys. Rev. B*, 68, 2003.